软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2011,22(7):1524−1537 [doi: 10.3724/SP.J.1001.2011.03869]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel/Fax: +86-10-62562563

可判定的时序动态描述逻辑*

常 亮[1+], 史忠植[2], 古天龙[1], 王晓峰[2]

[1](桂林电子科技大学 广西可信软件重点实验室,广西 桂林 541004)
[2](中国科学院 计算技术研究所 智能信息处理重点实验室,北京 100190)

Decidable Temporal Dynamic Description Logic

CHANG Liang[1+], SHI Zhong-Zhi[2], GU Tian-Long[1], WANG Xiao-Feng[2]

[1](Guangxi Key Laboratory of Trusted Software, Guilin University of Electronic Technology, Guilin 541004, China)
[2](Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, The Chinese Academy of Sciences, Beijing 100190)
+ Corresponding author: E-mail: changl@ics.ict.ac.cn

Chang L, Shi ZZ, Gu TL, Wang XF. Decidable temporal dynamic description logic. *Journal of Software*, 2011,22(7):1524−1537. http://www.jos.org.cn/1000-9825/3869.htm

Abstract: The dynamic description logic DDL (dynamic description logic) provides a kind of action theory based on description logics. It is a useful representation of the dynamic application domains in the environment of the Semantic Web. In order to bring the representation capability of the branching temporal logic into the dynamic description logic, this paper treats the time slices of temporal logics as the executions of atomic actions, so that the temporal dimension and the dynamic dimension can be unified. Based on this idea, constructed over the description logic ALCQIO, a temporal dynamic description logic, named TD_{ALCQIO}, is presented. Tableau decision algorithm is provided for TD_{ALCQIO}. Both the termination and the correctness of this algorithm have been proved. The logic TD_{ALCQIO} not only inherits the representation capability provided by the dynamic description logic constructed over ALCQIO (attributive language with complements, qualified number restrictions, inverse roles and nominals), but it also has the ability to describe and reason about some temporal features such as the reachability property and the safety property of the whole dynamic application domains. Therefore, TD_{ALCQIO} provides further support for knowledge representation and reasoning in the environment of the Semantic Web.
Key words: dynamic description logic; branching temporal logic; knowledge representation and reasoning; action theory; Tableau decision algorithm

摘 要: 动态描述逻辑 DDL(dynamic description logic)提供了一种基于描述逻辑的动作理论,适用于语义 Web 环境下对动态领域知识的刻画和推理.为了将分支时序逻辑的刻画能力引入到动态描述逻辑中,将时间的进展体现为原子动作的执行,从而将时序维与动态维统一起来.在此基础上,从描述逻辑 ALCQIO 出发构建了一个时序动态描述逻辑 TD_{ALCQIO},给出了 TD_{ALCQIO} 的 Tableau 判定算法,并证明了算法的可终止性和正确性.TD_{ALCQIO} 不仅兼容了构建在描述逻辑 ALCQIO 基础上的动态描述逻辑的刻画和推理能力,而且还可从可达性、安全性等角度对整个动态领域的时序特征进行刻画和推理,从而为语义 Web 环境下对动态领域知识的刻画和推理提供了进一步的逻辑支持.

* 基金项目: 国家自然科学基金(60903079, 60775035, 60963010, 60803033); 国家高技术研究发展计划(863)(2007AA01Z132); 国家重点基础研究发展计划(973)(2007CB311004); 广西自然科学基金(0832006Z)
收稿时间: 2009-11-03; 修改时间: 2010-03-05; 定稿时间: 2010-05-05

关键词: 动态描述逻辑;分支时序逻辑;知识表示和推理;动作理论;Tableau 判定算法
中图法分类号: TP181　　**文献标识码:** A

描述逻辑在语义 Web 中扮演着关键角色,是 W3C 推荐的 Web 本体语言 OWL(Web ontology language)的逻辑基础.描述逻辑的主要特点在于提供了命题逻辑所无法比拟的刻画能力,同时又保证了相关推理问题的可判定性,并且具有有效的判定算法和推理机制作为支撑.针对 Web 环境的各种特征和应用需求,研究者提出描述逻辑的各种扩展形式,以期为语义 Web 提供更为充分的逻辑支持.例如,针对 Web 环境的开放性提出了非单调描述逻辑[1]和多值描述逻辑[2],针对 Web 应用中需要处理的不确定信息提出了模糊描述逻辑和粗糙描述逻辑等[3,4].

针对描述逻辑只能处理静态领域知识的局限,文献[5]将动态逻辑的刻画成分引入到描述逻辑中,构建了一种动态描述逻辑 DDL(dynamic description logic).文献[6]针对描述逻辑 ALCO(attributive language with complements and nominals),ALCQO(attributive language with complements, qualified number restrictions and nominals)和 ALCQIO(attributive language with complements, qualified number restrictions, inverse roles and nominals)依次给出了相应的动态描述逻辑系统及其 Tableau 判定算法.文献[7]在 DDL 中引入了对动作执行过程进行刻画和推理的机制.DDL 提供了一类基于描述逻辑的动作理论,为语义 Web 服务的建模和推理以及在此基础上的服务发现和组合等提供了有效的途径和工具[8].

与动态逻辑相对应,时序逻辑是研究动态系统的另外一种有效工具.在对动态系统进行刻画时,动态逻辑主要关注于各个动作的输入与输出、前提条件与执行效果等,而时序逻辑则关注于整个动态系统所表现出来的时序特征.因此,在对描述逻辑进行动态扩展的基础上,如果进一步引入时序逻辑的刻画成分,则将为动态领域的知识刻画和推理提供更强的逻辑支持.但存在的挑战是:动态描述逻辑已经在描述逻辑的基础上引入了一个动态维,形成了具有二维特征的逻辑系统.如果再引入一个时序维,则可能会得到具有三维特征的逻辑系统,而具有三维特征的逻辑系统一般来说是不可判定的[9].

针对上述问题,本文借鉴了动态线性时序逻辑 DLTL(dynamic linear time temporal logic)中对时间的处理方式[10],将时间的进展体现为原子行动的执行,从而将动态维与时序维统一起来.在此基础上,本文在动态描述逻辑中引入分支时序逻辑的刻画成分,将动作执行过程中蕴含的各种时序关系通过路径量词和时态算子刻画出来.最终,本文在描述逻辑 ALCQIO 的基础上构建了一个时序动态描述逻辑 TD_{ALCQIO},并为其提供了相应的 Tableau 判定算法.

1　时序动态描述逻辑 TD_{ALCQIO}

TD_{ALCQIO} 的基本符号包括由概念名组成的集合 N_C、由角色名组成的集合 N_R、由个体名组成的集合 N_I 以及由原子动作名组成的集合 N_A.从这些符号出发,可以通过各种构造符递归地生成角色、概念、公式和动作.

定义 1. R 是 TD_{ALCQIO} 中的角色当且仅当 $R\in N_R\cup\{R_i^-|R_i\in N_R\}$.

定义 2. TD_{ALCQIO} 中的概念由如下产生式生成:

$$C,D::=C_i|\{u\}|\neg C|C\sqcup D|\forall R.C|\leqslant nR.C,$$

其中,$C_i\in N_C,u\in N_I,R$ 为角色,n 为非负整数.此外,与描述逻辑 ALCQIO 相同,可以引入形如 $C\sqcap D,\exists R.C,\geqslant nR.C,\top$ 以及 $\bot$ 的概念,分别作为 $\neg(\neg C\sqcup\neg D),\neg(\forall R.\neg C),\neg(\leqslant(n-1)R.C),C\sqcup\neg C$ 和 $\neg(C\sqcup\neg C)$ 的缩写.

令 C_i 为概念名,D 为概念,则称 $C_i\equiv D$ 为概念定义式.对于由概念定义式组成的任一有限集合 $\mathcal{T}$,如果每个概念名最多在 $\mathcal{T}$ 中概念定义式的左边出现 1 次,则称 $\mathcal{T}$ 为 TBox.

相对于某个 TBox $\mathcal{T}$,如果某个概念名 C_i 出现在 $\mathcal{T}$ 中概念定义式的左边,则称 C_i 为被定义的概念名,否则称其为简单概念名.

定义 3. TD_{ALCQIO} 中的公式由如下产生式生成:

$$\varphi,\psi::=C(u)|R(u,v)|\langle\pi\rangle\varphi|\boldsymbol{E}(\varphi\mathcal{U}^{\pi}\psi)|\boldsymbol{A}(\varphi\mathcal{U}^{\pi}\psi)|\neg\varphi|\varphi\vee\psi,$$

其中,$u,v\in N_I$,C 为概念,R 为角色,π为动作.此外,与动态描述逻辑 D-ALCQIO[6]相同,可以引入形如$[\pi]\varphi,\varphi\wedge\psi$, $\varphi\rightarrow\psi$,true 以及 false 的公式,分别作为$\neg\langle\pi\rangle\neg\varphi,\neg(\neg\varphi\vee\neg\psi),\neg\varphi\vee\psi,\varphi\vee\neg\varphi$和$\neg(\varphi\vee\neg\varphi)$的缩写.

给定某个 TBox $\mathcal{T}$,如果 C 为简单概念名,$u,v\in N_I$,R 为角色,则将形如 $C(u),\neg C(u),R(u,v)$以及$\neg R(u,v)$的公式都称为简单公式.

定义 4. TD_{ALCQIO} 中的动作由如下产生式生成:

$$\pi,\pi'::=\alpha|\varphi?|\pi\cup\pi'|\pi,\pi'|\pi^*,$$

其中,$\alpha\in N_A$,φ为公式.

相对于某个 TBox $\mathcal{T}$,将形如$\alpha\equiv(P,E)$的表达式称为原子动作定义式,其中:(1) $\alpha\in N_A$ 表示所定义的原子动作; (2) P 是由公式组成的有限集合,称为该原子动作的前提条件;(3) E 是由简单公式组成的有限集合,称为该原子动作的执行效果.

对于由原子动作定义式组成的任一有限集合 $\mathcal{A}c$,如果每个原子动作名最多在 $\mathcal{A}c$ 中某个定义式的左边出现 1 次,则称 $\mathcal{A}c$ 为 ActBox.

相对于某个 ActBox $\mathcal{A}c$,如果原子动作α出现在 $\mathcal{A}c$ 中某个定义式的左边,则称α是相对于 $\mathcal{A}c$ 被定义的.如果公式φ中出现的所有原子动作都是相对于 $\mathcal{A}c$ 被定义的,则称公式φ是相对于 $\mathcal{A}c$ 被定义的.

TD_{ALCQIO} 的语义结构从整体上体现为由多个可能世界构成的空间.在其中的每个可能世界下都分别对概念名、角色名和个体名进行解释;每个动作被解释为由关于可能世界的序列组成的集合,体现了动作执行过程中可能形成的所有轨迹.形式定义如下:

定义 5. TD_{ALCQIO} 结构是一个三元组 $M=(\Sigma,\Delta_M,I)$,其中:

(1) Σ是形如 $\Sigma=(W,\cdot^T)$的框架,在该框架中:W是由可能世界组成的非空集合,函数$\cdot^T$将 N_A 中的每个原子动作名α_i映射为 W 上的某个二元关系 $\alpha_i^T\subseteq W\times W$;

(2) Δ_M 是由个体组成的非空集合,作为该结构的论域;

(3) 函数 I 对 W 中每个可能世界 w 赋予一个解释 $I(w)=(\Delta_M,\cdot^{I(w)})$,其中的解释函数$\cdot^{I(w)}$满足以下条件:

(i) 将 N_C 中的每个概念名 C_i 解释为Δ_M 的某个子集 $C_i^{I(w)}\subseteq\Delta_M$;

(ii) 将 N_R 中的每个角色名 R_i 解释为Δ_M 上的某个二元关系 $R_i^{I(w)}\subseteq\Delta_M\times\Delta_M$;

(iii) 将 N_I 中的每个个体名 p_i 解释为Δ_M 中的某个元素 $p_i^{I(w)}\in\Delta_M$,并且对于 W 中任一可能世界 w'都有 $p_i^{I(w)}=p_i^{I(w')}$. 由于 p_i 的解释与可能世界无关,因而也将 $p_i^{I(w)}$ 简记为 p_i^I.

定义 6. 给定某个结构 $M=(\Sigma,\Delta_M,I)$以及由该结构中的可能世界构成的任一序列$(w_1,w_2,...)$,如果对于每一对可能世界 w_i 和 w_{i+1}(其中 $i\geqslant 1$)都存在某个$\alpha_i\in N_A$ 使得$(w_i,w_{i+1})\in\alpha_i^T$, 则称该序列是 M 中的一条轨迹.对于任一轨迹τ,分别用$|\tau|$和$\tau[i]$表示τ的长度以及τ中的第 i 个可能世界.

定义 7. 令 T_1,T_2 是由轨迹组成的两个集合,对它们的混合积 $T_1\times T_2$ 定义如下:

$$T_1\times T_2:=\{(\tau_1[1],\tau_1[2],\ldots,\tau_1[|\tau_1|],\tau_2[2],\ldots,\tau_2[|\tau_2|])|\tau_1\in T_1,\tau_2\in T_2,\tau_1\text{ 的长度有限并且 }\tau_1[|\tau_1|]=\tau_2[1]\},$$

即,对于分别来自 T_1 和 T_2 的每一对轨迹τ_1,τ_2,如果τ_1 的长度有限并且τ_1 的最后一个可能世界与τ_2 的第 1 个可能世界相同,则将这两条轨迹串联起来并且去掉其中重复的可能世界$\tau_2[1]$.

定义 8. 令 $M=(\Sigma,\Delta_M,I)$为 TD_{ALCQIO} 结构,其中,$\Sigma=(W,\cdot^T)$.对 TD_{ALCQIO} 中的概念、公式和动作的语义归纳定义如下.首先,对于 W 中的任一可能世界 w,将任一角色 R 解释为Δ_M 上的某个二元关系 $R^{I(w)}$,将任一概念 C 解释为Δ_M 的某个子集 $C^{I(w)}$.归纳定义如下:

(1) $(R^-)^{I(w)}:=\{(y,x)|(x,y)\in R^{I(w)}\}$;

(2) $\{u\}^{I(w)}:=\{u^I\}$,其中,$u\in N_I$;

(3) $(\neg C)^{I(w)}:=\Delta_M\backslash C^{I(w)}$,其中的“\”为集合差运算;

(4) $(C\sqcup D)^{I(w)}:=C^{I(w)}\cup D^{I(w)}$,其中的“∪”为集合并运算;

(5) $(\forall R.C)^{I(w)}:=\{x|$对于任一 $y\in\Delta_M$:如果$(x,y)\in R^{I(w)}$,则必然有 $y\in C^{I(w)}\}$;

(6) $(\leqslant nR.C)^{I(w)}:=\{x|\#\{y|(x,y)\in R^{I(w)}$并且$y\in C^{I(w)}\}\leqslant n\}$,其中的“#”表示集合的元素个数.

其次,对于 W 中的任一可能世界 w,用$(M,w)\vDash\varphi$表示公式φ在结构 M 中的可能世界 w 下成立.根据φ的结构归纳定义如下:

(7) $(M,w)\vDash C(u)$ iff $u^I\in C^{I(w)}$;

(8) $(M,w)\vDash R(u,v)$ iff $(u^I,v^I)\in R^{I(w)}$;

(9) $(M,w)\vDash\neg\varphi$ iff $(M,w)\nvDash\varphi$;

(10) $(M,w)\vDash\varphi\vee\psi$ iff $(M,w)\vDash\varphi$或者$(M,w)\vDash\psi$;

(11) $(M,w)\vDash\langle\pi\rangle\varphi$ iff 存在 M 中某条长度有限的轨迹τ,使得$\tau\in\pi^T,\tau[1]=w$ 和$(M,\tau[|\tau|])\vDash\psi$;

(12) $(M,w)\vDash\boldsymbol{E}(\varphi\mathcal{U}^{\pi}\psi)$ iff 存在 M 中的某条轨迹τ以及某个正整数 k,使得$\tau\in\pi^T,\tau[1]=w,1\leqslant k\leqslant|\tau|,(M,\tau[k])\vDash\psi$,并且对于任意 $1\leqslant i<k$ 都有$(M,\tau[i])\vDash\varphi$;

(13) $(M,w)\vDash\boldsymbol{A}(\varphi\mathcal{U}^{\pi}\psi)$ iff 对于 M 中的任一轨迹τ:如果$\tau\in\pi^T$并且$\tau[1]=w$,则必然存在某个正整数 k 使得 $1\leqslant k\leqslant|\tau|,(M,\tau[k])\vDash\psi$ 以及对于任意 $1\leqslant i<k$ 都有$(M,\tau[i])\vDash\varphi$.

最后,对于任一动作π,将其映射为由 M 中的轨迹组成的某个集合π^T.根据π的结构归纳定义如下:

(14) $(\varphi?)^T:=\{(w,w)|(M,w)\vDash\varphi\}$;

(15) $(\pi\cup\pi')^T:=\pi^T\cup\pi'^T$;

(16) $(\pi;\pi')^T:=\pi^T\times\pi'^T$,其中的“×”为关于轨迹集合的混合积运算;

(17) $(\pi^*)^T:=(\pi^0)^T\cup(\pi^1)^T\cup(\pi^2)^T\cup\ldots$,其中,$(\pi^0)^T:=\{(w,w)|w\in W\}$,并且对于任意 $i\geqslant1$ 都有$(\pi^i)^T:=(\pi^{i-1};\pi)^T$.

称某个 TD_{ALCQIO} 结构 $M=(\Sigma,\Delta_M,I)$(其中,$\Sigma=(W,\cdot^T)$)是某个 TBox $\mathcal{T}$ 的模型,表示为 $M\vDash\mathcal{T}$,当且仅当对于任一$C_i\equiv D\in\mathcal{T}$和任一 $w\in W$ 都有 $C_i^{I(w)}=D^{I(w)}$.

称某个 TD_{ALCQIO} 结构 $M=(\Sigma,\Delta_M,I)$(其中,$\Sigma=(W,\cdot^T)$)是某个 ActBox $\mathcal{A}c$ 的模型,表示为 $M\vDash\mathcal{A}c$,当且仅当对于任一$\alpha\equiv(P,E)\in\mathcal{A}c$ 都有$\alpha^T=\{(w,w')\in W\times W|$① 对于任一$\psi\in P$ 都有$(M,w)\vDash\psi$,② 对于任一简单概念名 C 都有 $C^{I(w')}=(C^{I(w)}\cup\{u^I|C(u)\in E\})\backslash\{u^I|\neg C(u)\in E\}$;以及③ 对于任一角色名 R 都有 $R^{I(w')}=(R^{I(w)}\cup\{(u^I,v^I)|R(u,v)\in E\})\backslash\{(u^I,v^I)|\neg R(u,v)\in E\}\}$.

公式的可满足性问题是 TD_{ALCQIO} 中最基本的推理问题,定义如下:

定义 9. 令 $\mathcal{T},\mathcal{A}c$ 分别为 TBox 和 ActBox,φ为公式,称φ相对于 $\mathcal{T}$ 和 $\mathcal{A}c$ 是可满足的,当且仅当存在某个结构 $M=(\Sigma,\Delta_M,I)$以及该结构中的某个可能世界 w 使得 $M\vDash\mathcal{T},M\vDash\mathcal{A}c$ 和$(M,w)\vDash\varphi$.

2 TD_{ALCQIO} 的描述和推理能力

从上一节关于概念、公式、动作、TBox 以及 ActBox 的语法和语义定义可以看出,TD_{ALCQIO} 兼容了构建在描述逻辑 ALCQIO 上的动态描述逻辑 D-ALCQIO[6]的描述能力.

此外,由于 TD_{ALCQIO} 中引入了形如$\boldsymbol{E}(\varphi\mathcal{U}^{\pi}\psi)$和$\boldsymbol{A}(\varphi\mathcal{U}^{\pi}\psi)$的公式,使其进一步具有了分支时序逻辑的刻画能力.具体来说,我们可以引入路径量词 $\boldsymbol{E},\boldsymbol{A}$ 和时态算子 $\boldsymbol{X},\mathcal{U}$,构造出形如 $\boldsymbol{EX}\ \varphi,\boldsymbol{E}(\varphi\mathcal{U}\psi)$和 $\boldsymbol{A}(\varphi\mathcal{U}\psi)$的时序断言,并将它们定义为 TD_{ALCQIO} 中已有公式的缩写形式,即:

$$\boldsymbol{EX}\ \varphi\triangleq\langle\alpha_1\cup\ldots\cup\alpha_n\rangle\varphi$$

$$\boldsymbol{E}(\varphi\mathcal{U}\psi)\triangleq\boldsymbol{E}(\varphi\mathcal{U}^{(\alpha_1\cup\ldots\cup\alpha_n)^*}\psi)$$

$$\boldsymbol{A}(\varphi\mathcal{U}\psi)\triangleq\boldsymbol{A}(\varphi\mathcal{U}^{(\alpha_1\cup\ldots\cup\alpha_n)^*}\psi)$$

其中的$\alpha_1,\ldots,\alpha_n$是 N_A 中所有的原子动作.将这些缩写带入定义 8 之后,可以得到如下语义解释:

(1) $(M,w)\vDash\boldsymbol{EX}\varphi$ iff 存在某个原子动作$\alpha\in N_A$和某个可能世界 $w'\in W$ 使得$(w,w')\in\alpha^T$并且$(M,w')\vDash\psi$;

(2) $(M,w)\vDash\boldsymbol{E}(\varphi\mathcal{U}\psi)$ iff 存在 M 中的某条轨迹τ以及某个正整数 k,使得$\tau[1]=w,1\leqslant k\leqslant|\tau|,(M,\tau[k])\vDash\psi$ 以及对于任意 $1\leqslant i<k$ 都有$(M,\tau[i])\vDash\varphi$;

(3) $(M,w)\vDash\boldsymbol{A}(\varphi\mathcal{U}\psi)$ iff 对于 M 中的任一轨迹τ:如果$\tau[1]=w$,则必然存在某个正整数 k 使得 $1\leqslant k\leqslant|\tau|$,

$(M,\tau[k])\vDash\psi$ 以及对于任意 $1\leqslant i<k$ 都有 $(M,\tau[i])\vDash\varphi$.

可见,时序断言 $\boldsymbol{EX}\ \varphi$,$\boldsymbol{E}(\varphi\mathcal{U}\psi)$和 $\boldsymbol{A}(\varphi\mathcal{U}\psi)$在 TD_{ALCQIO} 中的语义解释与在 CTL 中的语义解释[11]类似;区别在于 CTL 语义结构中无限长的状态序列在这里被限制成与具体动作相关的轨迹.在这些断言的基础上,我们还可以进一步引入时态算子 $\boldsymbol{F}$,$\boldsymbol{G}$,构造出形如 $\boldsymbol{EF}\ \varphi$,$\boldsymbol{AF}\ \varphi$,$\boldsymbol{EG}\ \varphi$,$\boldsymbol{AG}\ \varphi$和 $\boldsymbol{AX}\ \varphi$的时序断言,分别作为 $\boldsymbol{E}(\text{true}\ \mathcal{U}\ \varphi)$,$\boldsymbol{A}(\text{true}\ \mathcal{U}\ \varphi)$,$\neg\boldsymbol{AF}(\neg\varphi)$,$\neg\boldsymbol{EF}(\neg\varphi)$以及$\neg\boldsymbol{EX}(\neg\varphi)$的缩写.

下面通过例子进一步考察 TD_{ALCQIO} 的刻画和推理能力.假设某个 Web 服务系统可以让用户通过信用卡在线购买图书和 CD,我们应用 TD_{ALCQIO} 对其刻画和推理如下:

首先给出该系统中涉及的基本符号:

- 概念名集合 $N_C=\{person,creditCard,cd,book,customer,VIPCustomer,instore\}$;
- 角色名集合 $N_R=\{has,bought\}$;
- N_I 是由个体名 *Tom*,*Jack*,*Visa*,*HarryPotter*,*BackStreetBoys* 等组成的有限集合;
- N_A 是由原子动作名 $buyBook_{Tom,Har}$,$buyBook_{Jack,Har}$,$buyCD_{Tom,Bac}$,$buyCD_{Jack,Bac}$,$order_{Har}$,$order_{Bac}$ 等组成的有限集合.

在此基础上,可以将该系统中相关的静态领域知识刻画为如下概念定义式:

$$customer\equiv person\sqcap\exists has.creditCard,$$

$$VIPCustomer\equiv customer\sqcap\geqslant 3\ bought.(cd\sqcup book).$$

这些概念定义式的直观含义为:顾客是持有信用卡的人员;VIP 顾客是购买了至少 3 本 CD 或图书的顾客.用 $\mathcal{T}_{shop}$ 表示由这些概念定义式组成的 TBox.

同时,可以将该系统中提供的原子 Web 服务刻画为以下原子动作定义式:

$$buyBook_{Tom,Har}\equiv(\{customer(Tom),book(HarryPotter),instore(HarryPotter)\},$$
$$\{\neg instore(HarryPotter),bought(Tom,HarryPotter)\}),$$
$$buyCD_{Jack,Bac}\equiv(\{customer(Jack),cd(BackStreetBoys),instore(BackStreetBoys)\},$$
$$\{\neg instore(BackStreetBoys),bought(Jack,BackStreetBoys)\}),$$
$$order_{Har}\equiv(\{(book\sqcup cd)(HarryPotter),\neg instore(HarryPotter),(\neg\exists bought^{-}.person)(HarryPotter)\},$$
$$\{instore(HarryPotter)\}).$$

这 3 个原子服务依次表示:顾客 *Tom* 购买图书 *HarryPotter*,顾客 *Jack* 购买 CD *BackStreetBoys*,以及商家在库存中没有图书或 CD *HarryPotter* 的情况下进行进货.其中,关于 $order_{Har}$ 的原子动作定义式表示:如果 *HarryPotter* 是一本图书或 CD,库存中没有 *HarryPotter*,并且 *HarryPotter* 没有被任何人买走,则相应的原子服务可以被执行,并且在执行之后产生的效果是使得 *HarryPotter* 存在于库存中.其他两个原子动作定义式的直观含义可以类似地得到.

类似地,对原子动作 $buyBook_{Jack,Har}$,$buyCD_{Tom,Bac}$,$order_{Bac}$ 等也可以进行相应的定义,分别表示顾客 *Jack* 购买图书 *HarryPotter*,顾客 *Tom* 购买 CD *BackStreetBoys*,以及商家对 *BackStreetBoys* 进行进货.用 $\mathcal{A}c_{shop}$ 表示由上述原子动作定义式组成的 ActBox.

在原子动作定义式的基础上,还可以通过顺序、测试、选择、迭代等动作构造符对 Web 服务组合方案进行刻画.例如,该 Web 服务系统可能为 VIP 顾客提供某个 Web 服务组合方案,用来保证 VIP 顾客能够买到库存中没有的商品.其中,针对顾客 *Tom* 和图书 *HarryPotter*,可以将相应的组合方案 $VIPbuy_{Tom,Har}$ 刻画为

$$VIPCustomer(Tom)?;((instore(HarryPotter)?;buyBook_{Tom,Har})\cup(\neg instore(HarryPotter)?;order_{Har};buyBook_{Tom,Har})).$$

该组合方案首先要求 $VIPCustomer(Tom)$成立.接下来,如果 $instore(HarryPotter)$成立,则直接调用 Web 服务 $buyBook_{Tom,Har}$;否则,先调用 Web 服务 $order_{Har}$ 然后再调用 $buyBook_{Tom,Har}$.通过与动态描述逻辑相同的方式,Web 服务组合方案中经常用到的 Sequence,Choice,Any-Order,If-Then-Else,Iterate,Repeat-While,Repeat-Until 等控制结构都可以在 TD_{ALCQIO} 中得到刻画[8].

最后,可以通过一些不含有动作的公式对 Web 服务系统的当前状态进行刻画,例如,

$$VIPCustomer(Tom),person(Jack),creditCard(Visa),has(Jack,Visa),$$
$$book(HarryPotter),cd(BackStreetBoys),\neg instore(HarryPotter),instore(BackStreetBoys).$$

这些公式表达了以下信息:*Tom* 是一名 VIP 顾客,*Jack* 是一名持有 Visa 信用卡的人员,*HarryPotter* 是一本没在库存中的图书,*BackStreetBoys* 是一张在库存中的 CD.用 $\mathcal{A}_{\mathrm{shop}}$ 表示由这些公式组成的集合.

上面刻画的各部分知识一起形成了一个基于描述逻辑 ALCQIO 的动作理论[8],实现了对我们所考察的 Web 服务系统的建模.接下来,我们可以分别从动态特征和时序特征两个方面对该系统的性质进行考察.

一方面,借助形如$\langle\pi\rangle\varphi$和$[\pi]\ \varphi$的动作断言,可以对动作的可执行性问题、投影问题、规划问题等进行刻画.在此基础上,与文献[12]类似,可以借助公式的可满足性问题实现对这些问题的推理.

例如,公式$\langle VIPbuy_{Tom,Har}\rangle$true 的直观含义是"动作 $VIPbuy_{Tom,Har}$ 可以被成功地执行".如果该公式可以按照文献[12]的方式从上面构造的动作理论推导出来,则表明动作 $VIPbuy_{Tom,Har}$ 在 Web 服务系统的当前状态下是可以被执行的.

再如,公式$[VIPbuy_{Tom,Har}]bought(Tom,HarryPotter)$的直观含义是"只要动作 $VIPbuy_{Tom,Har}$ 执行结束,公式 $bought(Tom,HarryPotter)$就一定成立".同样,如果该公式可以从上面构造的动作理论推导出来,则表明 Web 服务系统在当前状态下具有相应的投影性质.

最后,如果上面两个公式所刻画的性质都成立,则表明动作 $VIPbuy_{Tom,Har}$ 是 Web 服务系统在当前状态下实现目标公式 $bought(Tom,HarryPotter)$的一个规划.类似地还可以验证,动作"$order_{Har};buyBook_{Tom,Har}$"也是在当前状态下实现目标公式 $bought(Tom,HarryPotter)$的一个规划.

另一方面,借助时序动作断言 $\boldsymbol{E}(\varphi\,\mathcal{U}^{\pi}\psi),\boldsymbol{A}(\varphi\,\mathcal{U}^{\pi}\psi)$以及在其基础上定义的 $\boldsymbol{EF}\ \varphi,\boldsymbol{E}(\varphi\,\mathcal{U}\psi),\boldsymbol{AG}\ \varphi$等时序断言,可以对 Web 服务系统的可达性和安全性进行刻画和推理.

可达性是指系统在运行过程中可以到达某个状态或出现某种情况.

例如:公式 $\boldsymbol{EF}\ bought(Tom,HarryPotter)$表示 Web 服务系统存在某种执行方式使得 *Tom* 买到 *HarryPotter*;公式 $\boldsymbol{E}((\neg\exists bought^{-}.person)(HarryPotter)\ \mathcal{U}\ bought(Tom,HarryPotter))$表示存在某种执行方式使得 *Tom* 买到 *HarryPotter*,且在此之前 *HarryPotter* 没有被任何人购买过;公式 $\boldsymbol{AG}(\boldsymbol{EF}(\exists bought^{-}.customer)(HarryPotter))$表示 Web 服务系统总可以达到让 *HarryPotter* 被顾客买走的情况.此外,借助时序动作断言 $\boldsymbol{E}(\varphi\,\mathcal{U}^{\pi}\psi)$,还可以针对某个具体的动作$\pi$来考察相应的可达性.例如,公式 $\boldsymbol{E}(\neg instore(HarryPotter)\ \mathcal{U}^{VIPbuy_{Tom,Har}}\ instore(HarryPotter))$表示存在动作 $VIPbuy_{Tom,Har}$ 的某种执行方式,使得在执行过程中会出现库存中有 *HarryPotter* 的情况,并且在此之前的库存中一直没有 *HarryPotter*.

为了借助公式可满足性问题来实现对可达性的推理,我们需要引入一个前提公式Σ来保证 Web 服务系统中的各个原子服务在前提条件得到满足的情况下都将被执行.具体来说,对于上面刻画的 Web 服务系统,令 $\alpha_1,\ldots,\alpha_n$ 是 TBox $\mathcal{T}_{\mathrm{shop}}$ 中定义的各个原子动作,令 $Conj(P_{a_n})$表示将原子动作α_i的前提条件 P_{a_n} 中所有元素合取后得到的合取式.那么相应的前提公式 Σ为$[(\alpha_1\cup\ldots\cup\alpha_n)^*]((Conj(P_{a_n})\rightarrow\langle\alpha_1\rangle\mathrm{true})\wedge\ldots\wedge(Conj(P_{a_n})\rightarrow\langle\alpha_n\rangle\mathrm{true}))$.在此基础上,对于任何一个关于可达性的公式$\varphi$来说,Web 服务系统具有相应的可达性特征当且仅当公式$\neg(\Sigma\rightarrow(Conj(\mathcal{A}_{\mathrm{shop}})\rightarrow\varphi))$相对于 TBox $\mathcal{T}_{\mathrm{shop}}$ 和 ActBox $\mathcal{A}c_{\mathrm{shop}}$ 是不可满足的.应用本文下一节给出的关于公式可满足性的判定算法,可以验证本文考察的 Web 服务系统具有前面列举的 3 个可达性特征.

安全性是指在系统运行过程中某些"不好的"情况永远不会发生.

例如:公式 $\boldsymbol{AG}\neg(bought(Jack,HarryPotter)\wedge bought(Tom,HarryPotter))$表示在任何时候图书 *HarryPotter* 都不会既被 *Jack* 购买了又被 *Tom* 购买了;公式 $\boldsymbol{AG}((\exists bought^{-}.customer)(BackStreetBoys)\vee instore(BackStreetBoys))$则表示在任何时候,CD *BackStreetBoys* 要么被顾客购买了,要么还在库存中.

对安全性的推理同样可以借助 TD_{ALCQIO} 中公式的可满足性问题来实现.具体来说,对于任何一个关于安全性的公式 ψ,Web 服务系统具有相应的安全性特征当且仅当公式$\neg(Conj(\mathcal{A}_{\mathrm{shop}})\rightarrow\psi)$相对于 TBox $\mathcal{T}_{\mathrm{shop}}$ 和 ActBox $\mathcal{A}c_{\mathrm{shop}}$ 是不可满足的.应用本文下一节给出的判定算法,可以验证这里的 Web 服务系统具有上面列举的两个安全

性特征.然而,假如我们将 Web 服务中的原子动作 $order_{Har}$ 改写为如下形式:

$$order_{Har}\equiv(\{(book\sqcup cd)(HarryPotter),\neg instore(HarryPotter)\},\{instore(HarryPotter)\}),$$

则可以验证,此时的 Web 服务系统将不再具有公式 $\boldsymbol{AG}\ \neg(bought(Jack,HarryPotter)\wedge bought(Tom,HarryPotter))$ 刻画的安全性质.

总之,除了具有动态描述逻辑中对动态领域进行刻画和推理的能力之外,TD_{ALCQIO} 还可以从可达性、安全性等角度对动态领域的整体特征进行刻画和推理.在对语义 Web 服务进行建模的基础上,这些刻画和推理能力将为 Web 服务的发现、组合以及对组合方案的验证等提供有效的支持[8].

3 TD_{ALCQIO} 的 Tableau 判定算法

令 $\mathcal{T},\mathcal{A}c$ 分别为 TBox 和 ActBox,φ_{init} 是相对于 $\mathcal{A}c$ 被定义的任一公式.本节给出 TD_{ALCQIO} 的 Tableau 判定算法,对 φ_{init} 相对于 $\mathcal{T}$ 和 $\mathcal{A}c$ 的可满足性进行判定.

判定算法的基本思路与文献[6,7]相同.为了表述简便,与文献[7]相同,我们首先引入符号 $nf_{\mathcal{T},\mathcal{A}c}(\varphi_{init})$,用来表示将 φ_{init} 中出现的每个被定义的概念名以及每个被定义的原子动作根据 $\mathcal{T}$ 和 $\mathcal{A}c$ 中的定义式进行替换后得到的公式,从而将 φ_{init} 中涉及到的与 $\mathcal{T}$ 和 $\mathcal{A}c$ 相关的信息都编码到 $nf_{\mathcal{T},\mathcal{A}c}(\varphi_{init})$ 中;由于我们要求 TBox $\mathcal{T}$ 和 ActBox $\mathcal{A}c$ 之间不存在循环定义[13],因而这种替换过程总是可以终止的.其次,对于由简单公式组成的任一集合 E,用 $E^{\neg}$ 表示集合 $\{\varphi^{\neg}|\varphi\in E\}$,其中的 $\varphi^{\neg}$ 表示与公式 $\neg\varphi$ 逻辑等价的简单公式.最后,对于任一个体名 u 和任一概念 C,引入形如 $@_uC$ 的概念;相应的语义定义为:对于任一 TD_{ALCQIO} 结构 $M=(\Sigma,\Delta_M,I)$ 以及其中的任一可能世界 w,如果 $u^I\in C^{I(w)}$,则 $(@_uC)^{I(w)}=\Delta_M$;否则,$(@_uC)^{I(w)}=\varnothing$.

在此基础上,对前缀以及带前缀的公式定义如下:

定义 10. 任一前缀 $\sigma.\varepsilon$ 是由动作 σ 以及关于简单公式的集合 ε 组成的二元组,并且满足如下产生式规则:

$$\sigma.\varepsilon\ ::=\ (\varnothing,\varnothing).\varnothing\ \ |\ \ \sigma;(P,E).(\varepsilon\backslash E^{\neg})\cup E,$$

其中,(P,E) 是表示成二元组形式的原子动作,$\sigma;(P,E)$ 是新生成的顺序动作,$(\varepsilon\backslash E^{\neg})\cup E$ 是新生成的由简单公式组成的集合.也将 $(\varnothing,\varnothing).\varnothing$ 称为初始前缀,在下文中表示为 $\sigma_0.\varepsilon_0$.

令 $\sigma.\varepsilon$ 为前缀,φ 为公式,则称 $\sigma.\varepsilon:\varphi$ 为带前缀的公式.

在动态描述逻辑 D-ALCQIO[6]和 EDDL(ALCQO)[7]的判定算法的基础上,本文算法的关键是对 TD_{ALCQIO} 中形如 $\boldsymbol{E}(\varphi\mathcal{U}^{\pi}\psi),\neg\boldsymbol{E}(\varphi\mathcal{U}^{\pi}\psi),\boldsymbol{A}(\varphi\mathcal{U}^{\pi}\psi)$ 和 $\neg\boldsymbol{A}(\varphi\mathcal{U}^{\pi}\psi)$ 的公式进行处理,难点在于对这些公式中动作 π 的执行过程进行监控.针对该难点,下面将这些公式改写为基于有穷自动机的表示形式.

首先,对于任一动作 π,令 Σ_π 是由出现在 π 中的所有原子动作和测试动作组成的有限集合.在此基础上,将 Σ_π 中的每个元素看作一个字符,用 $L(\pi)$ 表示通过以下规则定义的由字符串组成的最小集合:① 如果 α 为原子动作或测试动作,则 $L(\alpha):=\{\alpha\}$;② $L(\pi\cup\pi'):=L(\pi)\cup L(\pi')$;③ $L(\pi;\pi'):=\{l_1l_2|l_1\in L(\pi)$ 并且 $l_2\in L(\pi')\}$;④ $L(\pi^*):=L(\pi^0)\cup L(\pi^1)\cup L(\pi^2)\cup\ldots$,其中,$L(\pi^0)$ 为空串,并且对于任意 $i\geqslant 1$ 都有 $L(\pi^i):=L(\pi^{i-1};\pi)$.显然,$L(\pi)$ 中的各个字符串分别对应于动作 π 的一种执行情况;另一方面,动作 π 的所有执行情况都包括在 $L(\pi)$ 中.

对于任一字符串 $l\in L(\pi)$,用 $|l|$ 表示该字符串的长度,用 $l[i]$ 表示其中的第 i 个字母(即出现在第 i 个位置上的原子动作或测试动作).借助这些符号,可以将定义 8 中时序动作断言的语义定义改写为以下形式:

(12′) $(M,w)\vDash\boldsymbol{E}(\varphi\mathcal{U}^{\pi}\psi)$ iff 存在某个字符串 $l\in L(\pi,M$ 中的某条轨迹 τ 以及某个正整数 k,使得:$|\tau|=|l|+1$,对于任意 $1\leqslant i\leqslant|l|$ 都有 $(\tau[i],\tau[i+1])\in l[i]^T,\tau[1]=w,1\leqslant k\leqslant|\tau|,(M,\tau[k])\vDash\psi$ 以及对于任意 $1\leqslant j<k$ 都有 $(M,\tau[j])\vDash\varphi$;

(13′) $(M,w)\vDash\boldsymbol{A}(\varphi\mathcal{U}^{\pi}\psi)$ iff 对于任一字符串 $l\in L(\pi)$ 和 M 中的任一轨迹 τ.如果 $|\tau|=|l|+1,\tau[1]=w$ 以及对于任意 $1\leqslant i\leqslant|l|$ 都有 $(\tau[i],\tau[i+1])\in l[i]^T$,则必然存在某个正整数 k 使得 $1\leqslant k\leqslant|\tau|,(M,\tau[k])\vDash\psi$ 以及对于任意 $1\leqslant j<k$ 都有 $(M,\tau[j])\vDash\varphi$.

显然,上述定义分别与定义 8(12)和定义 8(13)等价,区别在于这里突出了引起状态转换的原子动作和测试动作.

其次,从形式语言的角度看,TD_{ALCQIO} 中对动作的语法定义满足形式语言中对正规表达式的要求;任一动作π都是字母表Σ_π上的正规表达式,相应的 $L(\pi)$则是由π表达的正规集.因此,基于形式语言与自动机理论,对任一动作π都可以相应地构造一个非确定有穷自动机 $A=(Q,\Sigma_\pi,\delta,q_0,F)$,使得该自动机接受的语言(用 $L(A)$表示)恰好为 $L(\pi)$;其中,Q 为有穷状态集,Σ_π是自动机 A 的输入字符表,$\delta:Q\times\Sigma_\pi\to 2^Q$ 为转移函数,$q_0\in Q$ 为初始状态,$F\subseteq Q$ 为终结状态集.该构造过程可以在多项式时间内完成,并且自动机 A 中的状态个数与动作π的长度呈线性关系[14].

例如,如果动作π为$\alpha_1;(\varphi?;(\alpha_2\cup\alpha_3))^*$,其中的$\alpha_1,\alpha_2$ 和α_3 为原子动作,则构造后得到的自动机为 $A=(Q,\Sigma_\pi,\delta,q_0,F)$,其中,$Q=\{q_0,q_1,q_2\}$,$\Sigma_\pi=\{\alpha_0,\alpha_1,\alpha_2,\varphi?\}$,$F=\{q_1\}$,并且有$\delta(q_0,\alpha_1)=q_1$,$\delta(q_1,\varphi?)=q_2$,$\delta(q_2,\alpha_2)=q_1$和$\delta(q_2,\alpha_3)=q_1$.

对于任一非确定有穷自动机 $A=(Q,\Sigma_\pi,\delta,q_0,F)$以及其中的任一状态 $q\in Q$,本文用 $A(q)$表示将 A 的初始状态置为 q 之后得到的状态机,即 $A(q)=(Q,\Sigma_\pi,\delta,q,F)$.

接下来,我们引入形如 $\mathbf{E}(\varphi\mathcal{U}^{A(q)}\psi)$和 $\mathbf{A}(\varphi\mathcal{U}^{A(q)}\psi)$的带自动机的公式,并且在定义 8 的基础上将它们的语义定义为以下形式:

(1) $(M,w)\vDash\mathbf{E}(\varphi\mathcal{U}^{A(q)}\psi)$ iff 存在某个字符串 $l\in L(A(q))$、M 中的某条轨迹τ以及某个正整数 k,使得:$|\tau|=|l|+1$,对于任意 $1\leqslant i\leqslant|l|$都有$(\tau[i],\tau[i+1])\in l[i]^T$,$\tau[1]=w$,$1\leqslant k\leqslant|\tau|$,$(M,\tau[k])\vDash\psi$ 以及对于任意 $1\leqslant j<k$ 都有$(M,\tau[j])\vDash\varphi$;

(2) $(M,w)\vDash\mathbf{A}(\varphi\mathcal{U}^{A(q)}\psi)$ iff 对于任一字符串 $l\in L(A(q))$以及 M 中的任一轨迹τ.如果$|\tau|=|l|+1$,$\tau[1]=w$ 并且对于任意 $1\leqslant i\leqslant|l|$都有$(\tau[i],\tau[i+1])\in l[i]^T$,则必然存在某个正整数 k 使得 $1\leqslant k\leqslant|\tau|$,$(M,\tau[k])\vDash\psi$ 以及对于任意 $1\leqslant j<k$ 都有$(M,\tau[j])\vDash\varphi$.

最后,从以上语义定义可以比较直接地得到以下性质:

性质 1. 对于任一动作π和任一非确定有穷自动机 $A(q)=(Q,\Sigma,\delta,q,F)$,如果 $L(\pi)=L(A(q))$,则公式 $\mathbf{E}(\varphi\mathcal{U}^{\pi}\psi)$和 $\mathbf{A}(\varphi\mathcal{U}^{\pi}\psi)$分别与公式 $\mathbf{E}(\varphi\mathcal{U}^{A(q)}\psi)$和 $\mathbf{A}(\varphi\mathcal{U}^{A(q)}\psi)$逻辑等价.

性质 2. 对于任一非确定有穷自动机 $A(q)=(Q,\Sigma,\delta,q,F)$,存在下面 3 组关于公式的逻辑等价关系:

(A1) $\mathbf{A}(\varphi\mathcal{U}^{A(q)}\psi)\equiv\psi\vee(\varphi\wedge\neg(\vee_{\alpha\in\Sigma,q'\in\delta(q,\alpha)}\langle\alpha\rangle\neg\mathbf{A}(\varphi\mathcal{U}^{A(q')}\psi)))$;

(A2) 如果 $q\in F$,则 $\mathbf{E}(\varphi\mathcal{U}^{A(q)}\psi)\equiv\psi\vee(\varphi\wedge(\vee_{\alpha\in\Sigma,q'\in\delta(q,\alpha)}\langle\alpha\rangle\mathbf{E}(\varphi\mathcal{U}^{A(q')}\psi)))$;

(A3) 如果 $q\notin F$,则 $\mathbf{E}(\varphi\mathcal{U}^{A(q)}\psi)\equiv(\psi\wedge\mathbf{E}(\text{true}\,\mathcal{U}^{A(q)}\text{true}))\vee(\varphi\wedge(\vee_{\alpha\in\Sigma,q'\in\delta(q,\alpha)}\langle\alpha\rangle\mathbf{E}(\varphi\mathcal{U}^{A(q')}\psi)))$.

需要指出的是,性质 2 中针对公式 $\mathbf{E}(\varphi\mathcal{U}^{A(q)}\psi)$分别考察了 $q\in F$ 和 $q\notin F$ 两种情况.其中,当 $q\notin F$ 时,如果当前状态下公式 ψ 已经成立,则还要求公式 $\mathbf{E}(\text{true}\,\mathcal{U}^{A(q)}\text{true})$也成立,目的是保证 $\mathbf{E}(\varphi\mathcal{U}^{A(q)}\psi)$语义定义中的“存在某个字符串 $l\in L(A(q))$和 M 中的某条轨迹τ,使得$|\tau|=|l|+1$ 并且对于任意 $1\leqslant i\leqslant|l|$都有$(\tau[i],\tau[i+1])\in l[i]^T$”.

与文献[6,7]相同,TD_{ALCQIO} 中 Tableau 判定算法的基本操作是应用扩展规则对分枝逐步进行扩展.下面先给出对分枝及相关术语的定义.

定义 11. 任一分枝 $\mathcal{B}$ 是由以下 4 类元素组成的有限集合:(1) 带前缀的公式;(2) 形如 $X\equiv\langle\pi^*\rangle\varphi$的可能性标记,其中的$\langle\pi^*\rangle\varphi$是由迭代动作$\pi^*$和公式$\varphi$构成的动作断言,$X$ 为符号串;(3) 形如 $u=v$ 的关于个体名的等式,其中,$u,v\in N_I$;(4) 形如 $u\neq v$ 的关于个体名的不等式,其中,$u,v\in N_I$.

对于任一分枝 $\mathcal{B}$,用 $\mathcal{B}_{\text{Ind}}$ 表示由 $\mathcal{B}$ 中关于个体名的所有等式和不等式组成的集合.在此基础上,与文献[7]相同,用 $\mathcal{B}_{\text{Ind}}^*$ 表示 $\mathcal{B}_{\text{Ind}}$ 关于等价关系“=”和对称关系“≠”的闭包.此外,对于任一个体名 u,用$[u]_{=\mathcal{B}}$表示 u 在 $\mathcal{B}$ 中关于等价关系“=”的等价类,即$[u]_{=\mathcal{B}}:=\{v|u=v\in\mathcal{B}_{\text{Ind}}^*\}$.

对于任一分枝 $\mathcal{B}$,如果满足表 1~表 5 中某条扩展规则的前件部分,则可以根据该规则相应地在 $\mathcal{B}$ 中加入带前缀的公式、关于个体名的等式或者关于个体名的不等式.

定义 12. 如果表 1~表 5 中所有扩展规则都不能对 $\mathcal{B}$ 进行扩展,则称 $\mathcal{B}$ 为饱和的.

定义 13. 对于任一$\sigma.\varepsilon{:}\mathbf{E}(\varphi\mathcal{U}^{A(q)}\psi)\in\mathcal{B}$(其中,$A(q)=(Q,\Sigma,\delta,q,F)$),如果存在某两个前缀$\sigma_1.\varepsilon_1,\sigma_2.\varepsilon_2$ 以及自动机 $A(q)$中的某个状态 q',使得$\varepsilon_1=\varepsilon_2$,$q'\in F$,$\sigma_1.\varepsilon_1{:}\mathbf{E}(\varphi\mathcal{U}^{A(q')}\psi)\in\mathcal{B}$ 和$\sigma_2.\varepsilon_2{:}\psi\in\mathcal{B}$,则称$\sigma.\varepsilon{:}\mathbf{E}(\varphi\mathcal{U}^{A(q)}\psi)$在 $\mathcal{B}$ 中被实现了;否则,称$\sigma.\varepsilon{:}\mathbf{E}(\varphi\mathcal{U}^{A(q)}\psi)$为在 $\mathcal{B}$ 中未被实现的带自动机的存在性时序动作断言.

对于某个可能性标记 $X\equiv\langle\pi^*\rangle\varphi$,如果存在某两个前缀$\sigma.\varepsilon$和$\sigma'.\varepsilon'$使得$\varepsilon=\varepsilon'$,$\sigma.\varepsilon:X\in\mathcal{B}$ 以及$\sigma'.\varepsilon':\neg\varphi\in\mathcal{B}$,则称$X\equiv\langle\pi^*\rangle\varphi$在 $\mathcal{B}$ 中被实现了;否则,将 $X\equiv\langle\pi^*\rangle\varphi$称为在 $\mathcal{B}$ 中未被实现的可能性标记.

对于某个已经饱和的分枝 $\mathcal{B}$,如果 $\mathcal{B}$ 中存在未被实现的可能性标记或者未被实现的带自动机的存在性时序动作断言,则称分枝 $\mathcal{B}$ 是可忽略的.

定义 14. 如果分枝 $\mathcal{B}$ 中至少出现以下某种情况,则称 $\mathcal{B}$ 存在冲突;否则,称 $\mathcal{B}$ 为无冲突的:(1) 存在某个公式φ以及某两个前缀$\sigma.\varepsilon$和$\sigma'.\varepsilon'$,使得$\varepsilon=\varepsilon'$,$\sigma.\varepsilon$: $\varphi\in\mathcal{B}$ 和$\sigma'.\varepsilon':\neg\varphi\in\mathcal{B}$;(2) 存在某个概念 C、某个个体名 u 以及某两个前缀$\sigma.\varepsilon$和$\sigma'.\varepsilon'$,使得$\varepsilon=\varepsilon'$,$\sigma.\varepsilon:C(u)\in\mathcal{B}$ 和$\sigma'.\varepsilon':(\neg C)(u)\in\mathcal{B}$;(3) 存在某个个体名 u 使得 $u\neq u\in\mathcal{B}^*_{\text{Ind}}$.

判定算法中用到的扩展规则分别见表 1~表 5.这些扩展规则比较直观地体现了 $\text{TD}_{\text{ALCQIO}}$ 中的语义定义.其中,表 1 在文献[7]的关于概念的扩展规则中增加了$\neg r^-$-规则和 r^--规则,用来对角色逆反算子进行处理;表 2~表 4 的扩展规则与文献[7]中针对 EDDL(ALCQO)的相应的扩展规则相同.表 5 给出了关于时序动作断言的扩展规则;其中的 A-规则直观地体现了性质 1,$A\mathcal{U}$-和$\neg A\mathcal{U}$-规则体现了性质 2 中的(A1),$E\mathcal{U}_f$-和$\neg E\mathcal{U}_f$-规则体现了性质 2 中的(A2),$E\mathcal{U}_{nf}$-和$\neg E\mathcal{U}_{nf}$-规则体现了性质 2 中的(A3).

最后可以给出 $\text{TD}_{\text{ALCQIO}}$ 的 Tableau 判定算法.

算法 1. 按以下步骤判断公式φ_{init}相对于 $\mathcal{T}$和 $\mathcal{A}c$ 是否可满足:

(1) 构造分枝 $\mathcal{B}_{init}:=\{\sigma_0.\varepsilon_0:nf_{\mathcal{T},\mathcal{A}c}(\varphi_{init})\}$;

(2) 以任意顺序应用表 1~表 5 的扩展规则对 $\mathcal{B}_{init}$ 进行扩展;

(3) 如果通过扩展后能够得到某个饱和的分枝 $\mathcal{B}$,并且 $\mathcal{B}$ 是无冲突的和不可忽略的,则返回结果“φ_{init} 相对于 $\mathcal{T}$和 $\mathcal{A}c$ 是可满足的”;否则,返回结果“φ_{init} 相对于 $\mathcal{T}$和 $\mathcal{A}c$ 是不可满足的”.

Table 1 Tableau expansion rules on $\text{TD}_{\text{ALCQIO}}$ roles and concepts

表 1 $\text{TD}_{\text{ALCQIO}}$ 中关于角色和概念的扩展规则

Name	Rule
$\neg r^-$-rule	If $\sigma.\varepsilon:\neg R^-(x,y)\in\mathcal{B}$ and $\sigma.\varepsilon:\neg R(y,x)\notin\mathcal{B}$, then set $\mathcal{B}:=\mathcal{B}\cup\{\sigma.\varepsilon:\neg R(y,x)\}$.
r^--rule	If $\sigma.\varepsilon:R^-(x,y)\in\mathcal{B}$ and $\sigma.\varepsilon:R(y,x)\notin\mathcal{B}$, then set $\mathcal{B}:=\mathcal{B}\cup\{\sigma.\varepsilon:R(y,x)\}$.
$\neg\neg_C$-rule	If $\sigma.\varepsilon:(\neg(\neg C))(x)\in\mathcal{B}$ and $\sigma.\varepsilon:C(x)\notin\mathcal{B}$, then set $\mathcal{B}:=\mathcal{B}\cup\{\sigma.\varepsilon:C(x)\}$.
$\{\}$-rule	If $\sigma.\varepsilon:\{u\}(x)\in\mathcal{B}$ and $x=u\notin\mathcal{B}^*_{\text{Ind}}$, then set $\mathcal{B}:=\mathcal{B}\cup\{x=u\}$.
$\neg\{\}$-rule	If $\sigma.\varepsilon:(\neg\{u\})(x)\in\mathcal{B}$ and $x\neq u\notin\mathcal{B}^*_{\text{Ind}}$, then set $\mathcal{B}:=\mathcal{B}\cup\{x\neq u\}$.
@-rule	If $\sigma.\varepsilon:(@_uC)(x)\in\mathcal{B}$ and $\sigma.\varepsilon:C(u)\notin\mathcal{B}$, then set $\mathcal{B}:=\mathcal{B}\cup\{\sigma.\varepsilon:C(u)\}$.
$\neg$@-rule	If $\sigma.\varepsilon:(\neg@_uC)(x)\in\mathcal{B}$ and $\sigma.\varepsilon:(\neg C)(u)\notin\mathcal{B}$, then set $\mathcal{B}:=\mathcal{B}\cup\{\sigma.\varepsilon:(\neg C)(u)\}$.
$\sqcup$-rule	If $\sigma.\varepsilon:(C_1\sqcup C_2)(x)\in\mathcal{B}$, $\sigma.\varepsilon:C_1(x)\notin\mathcal{B}$ and $\sigma.\varepsilon:C_2(x)\notin\mathcal{B}$, then set $\mathcal{B}:=\mathcal{B}\cup\{\sigma.\varepsilon:C(x)\}$ for some $C\in\{C_1,C_2\}$.
$\neg\sqcup$-rule	If $\sigma.\varepsilon:(\neg(C_1\sqcup C_2))(x)\in\mathcal{B}$, and $\sigma.\varepsilon:(\neg C_1)(x)\notin\mathcal{B}$ or $\sigma.\varepsilon:(\neg C_2)(x)\notin\mathcal{B}$, then set $\mathcal{B}:=\mathcal{B}\cup\{\sigma.\varepsilon:(\neg C_1)(x),\sigma.\varepsilon:(\neg C_2)(x)\}$.
$\forall$-rule	If $\sigma_0.\varepsilon_0:(\forall R.C)(x)\in\mathcal{B}$, $\sigma_0.\varepsilon_0:R(x,y)\in\mathcal{B}$ and $\sigma_0.\varepsilon_0:C(y)\notin\mathcal{B}$, then set $\mathcal{B}:=\mathcal{B}\cup\{\sigma_0.\varepsilon_0:C(y)\}$.
$\neg\forall$-rule	If $\sigma_0.\varepsilon_0:(\neg(\forall R.C))(x)\in\mathcal{B}$, and no individual name y exists with both $\sigma_0.\varepsilon_0:R(x,y)\in\mathcal{B}$ and $\sigma_0.\varepsilon_0:(\neg C)(y)\in\mathcal{B}$, then introduce a new individual name y, and set $\mathcal{B}:=\mathcal{B}\cup\{\sigma_0.\varepsilon_0:R(x,y),\sigma_0.\varepsilon_0:(\neg C)(y)\}$.
$\leqslant$-rule	If $\sigma_0.\varepsilon_0:(\leqslant nR.C)(x)\in\mathcal{B}$, and there are n+1 individual names $y_1,\dots,y_{n+1}$ such that ① $[y_i]_{=\mathcal{B}}\neq[y_j]_{=\mathcal{B}}$ for every $1\leqslant i<j\leqslant n+1$ and ② both $\sigma_0.\varepsilon_0:R(x,y_k)\in\mathcal{B}$ and $\sigma_0.\varepsilon_0:C(y_k)\in\mathcal{B}$ for every $1\leqslant k\leqslant n+1$, then select any two individual names y_i, y_j and set $\mathcal{B}:=\mathcal{B}\cup\{y_i=y_j\}$.
$\neg\leqslant$-rule	If $\sigma_0.\varepsilon_0:(\neg(\leqslant nR.C))(x)\in\mathcal{B}$, and there are not n+1 individual names $y_1,\dots,y_{n+1}$ with ① $[y_i]_{=\mathcal{B}}\neq[y_j]_{=\mathcal{B}}$ for every $1\leqslant i<j\leqslant n+1$ and ② both $\sigma_0.\varepsilon_0:R(x,y_k)\in\mathcal{B}$ and $\sigma_0.\varepsilon_0:C(y_k)\in\mathcal{B}$ for every $1\leqslant k\leqslant n+1$, then introduce n+1 new individual names $y_1,\dots,y_{n+1}$ and set $\mathcal{B}:=\mathcal{B}\cup\{\sigma_0.\varepsilon_0:R(x,y_k)\mid 1\leqslant k\leqslant n+1\}\cup\{\sigma_0.\varepsilon_0:C(y_k)\mid 1\leqslant k\leqslant n+1\}\cup\{y_i\neq y_j\mid 1\leqslant i<j\leqslant n+1\}$.

Table 2 Tableau expansion rules on $\text{TD}_{\text{ALCQIO}}$ formulas except action assertions and temporal action assertions

表 2 $\text{TD}_{\text{ALCQIO}}$ 中除动作断言和时序动作断言之外的关于公式的扩展规则

Name	Rule
$\neg_f$-rule	If $\sigma.\varepsilon:\neg(C(x))\in\mathcal{B}$ and $\sigma.\varepsilon:(\neg C)(x)\notin\mathcal{B}$, then set $\mathcal{B}:=\mathcal{B}\cup\{\sigma.\varepsilon:(\neg C)(x)\}$.
$\neg\neg_f$-rule	If $\sigma.\varepsilon:\neg(\neg\varphi)\in\mathcal{B}$ and $\sigma.\varepsilon:\varphi\notin\mathcal{B}$, then set $\mathcal{B}:=\mathcal{B}\cup\{\sigma.\varepsilon:\varphi\}$.
$\vee$-rule	If $\sigma.\varepsilon:\varphi\vee\psi\in\mathcal{B}$, $\sigma.\varepsilon:\varphi\notin\mathcal{B}$ and $\sigma.\varepsilon:\psi\notin\mathcal{B}$, then set $\mathcal{B}:=\mathcal{B}\cup\{\sigma.\varepsilon:\phi\}$ for some $\phi\in\{\varphi,\psi\}$.
$\neg\vee$-rule	If $\sigma.\varepsilon:\neg(\varphi\vee\psi)\in\mathcal{B}$, and $\sigma.\varepsilon:\neg\varphi\notin\mathcal{B}$ or $\sigma.\varepsilon:\neg\psi\notin\mathcal{B}$, then set $\mathcal{B}:=\mathcal{B}\cup\{\sigma.\varepsilon:\neg\varphi,\sigma.\varepsilon:\neg\psi\}$.

Table 3 Tableau expansion rules on action assertions of TD_{ALCQIO}

表 3 TD_{ALCQIO} 中关于动作断言的扩展规则

Name	Rule
$;_{\langle\rangle}$-rule	If $\sigma.\varepsilon:\langle\pi_1;\pi_2\rangle\varphi\in\mathcal{B}$ and $\sigma.\varepsilon:\langle\pi_1\rangle\langle\pi_2\rangle\varphi\notin\mathcal{B}$, then set $\mathcal{B}:=\mathcal{B}\cup\{\sigma.\varepsilon:\langle\pi_1\rangle\langle\pi_2\rangle\varphi\}$.
$\neg;_{\langle\rangle}$-rule	If $\sigma.\varepsilon:\neg\langle\pi_1;\pi_2\rangle\varphi\in\mathcal{B}$ and $\sigma.\varepsilon:\neg\langle\pi_1\rangle\langle\pi_2\rangle\varphi\notin\mathcal{B}$, then set $\mathcal{B}:=\mathcal{B}\cup\{\sigma.\varepsilon:\neg\langle\pi_1\rangle\langle\pi_2\rangle\varphi\}$.
$?_{\langle\rangle}$-rule	If $\sigma.\varepsilon:\langle\psi?\rangle\varphi\in\mathcal{B}$, and $\sigma.\varepsilon:\psi\notin\mathcal{B}$ or $\sigma.\varepsilon:\varphi\notin\mathcal{B}$, then set $\mathcal{B}:=\mathcal{B}\cup\{\sigma.\varepsilon:\psi,\sigma.\varepsilon:\varphi\}$.
$\neg?_{\langle\rangle}$-rule	If $\sigma.\varepsilon:\neg\langle\psi?\rangle\varphi\in\mathcal{B}$, $\sigma.\varepsilon:\neg\psi\notin\mathcal{B}$ and $\sigma.\varepsilon:\neg\varphi\notin\mathcal{B}$, then set $\mathcal{B}:=\mathcal{B}\cup\{\sigma.\varepsilon:\neg\psi\}$ or $\mathcal{B}:=\mathcal{B}\cup\{\sigma.\varepsilon:\neg\varphi\}$.
$\cup_{\langle\rangle}$-rule	If $\sigma.\varepsilon:\langle\pi_1\cup\pi_2\rangle\varphi\in\mathcal{B}$, $\sigma.\varepsilon:\langle\pi_1\rangle\varphi\notin\mathcal{B}$ and $\sigma.\varepsilon:\langle\pi_2\rangle\varphi\notin\mathcal{B}$, then set $\mathcal{B}:=\mathcal{B}\cup\{\sigma.\varepsilon:\langle\pi_1\rangle\varphi\}$ or $\mathcal{B}:=\mathcal{B}\cup\{\sigma.\varepsilon:\langle\pi_2\rangle\varphi\}$.
$\neg\cup_{\langle\rangle}$-rule	If $\sigma.\varepsilon:\neg\langle\pi_1\cup\pi_2\rangle\varphi\in\mathcal{B}$, and $\sigma.\varepsilon:\neg\langle\pi_1\rangle\varphi\notin\mathcal{B}$ or $\sigma.\varepsilon:\neg\langle\pi_2\rangle\varphi\notin\mathcal{B}$, then set $\mathcal{B}:=\mathcal{B}\cup\{\sigma.\varepsilon:\neg\langle\pi_1\rangle\varphi,\sigma.\varepsilon:\neg\langle\pi_2\rangle\varphi\}$.
${}^{*}_{\langle\rangle}$-rule	If $\sigma.\varepsilon:\langle\pi^{*}\rangle\varphi\in\mathcal{B}$, and there is no flag $X\equiv\langle\pi^{*}\rangle\varphi$ with $\sigma.\varepsilon:X\in\mathcal{B}$, then introduce a new string X and set $\mathcal{B}:=\mathcal{B}\cup\{X\equiv\langle\pi^{*}\rangle\varphi,\sigma.\varepsilon:X\}$.
X-rule	If there is some flag $X\equiv\langle\pi^{*}\rangle\varphi$ with $\sigma.\varepsilon:X\in\mathcal{B}$, $\sigma.\varepsilon:\varphi\notin\mathcal{B}$ and $\sigma.\varepsilon:\langle\pi\rangle X\notin\mathcal{B}$, then set $\mathcal{B}:=\mathcal{B}\cup\{\sigma.\varepsilon:\varphi\}$ or $\mathcal{B}:=\mathcal{B}\cup\{\sigma.\varepsilon:\neg\varphi,\sigma.\varepsilon:\langle\pi\rangle X\}$.
$\neg{}^{*}_{\langle\rangle}$-rule	If $\sigma.\varepsilon:\neg\langle\pi^{*}\rangle\varphi\in\mathcal{B}$, and $\sigma.\varepsilon:\neg\varphi\notin\mathcal{B}$ or $\sigma.\varepsilon:\neg\langle\pi\rangle\langle\pi^{*}\rangle\varphi\notin\mathcal{B}$, then set $\mathcal{B}:=\mathcal{B}\cup\{\sigma.\varepsilon:\neg\varphi,\sigma.\varepsilon:\neg\langle\pi\rangle\langle\pi^{*}\rangle\varphi\}$.
$atom_{\langle\rangle}$-rule	If $\sigma.\varepsilon:\langle(P,E)\rangle\varphi\in\mathcal{B}$, and $\{\sigma.\varepsilon:\psi\mid\psi\in P\}\not\subset\mathcal{B}$ or no prefix $\sigma'.\varepsilon'$ exists with both $\varepsilon'=(\varepsilon\backslash E^{\neg})\cup E$ and $\sigma'.\varepsilon':\varphi\in\mathcal{B}$, then: (Step 1) if $\{\sigma.\varepsilon:\psi\mid\psi\in P\}\not\subset\mathcal{B}$, then set $\mathcal{B}:=\mathcal{B}\cup\{\sigma.\varepsilon:\psi\mid\psi\in P\}$; and (Step 2) if there is no prefix $\sigma'.\varepsilon'$ with both $\varepsilon'=(\varepsilon\backslash E^{\neg})\cup E$ and $\sigma'.\varepsilon':\varphi\in\mathcal{B}$, then: if no prefix $\sigma'.\varepsilon'$ exists with $\varepsilon'=(\varepsilon\backslash E^{\neg})\cup E$, then introduce a prefix $\sigma'.\varepsilon':=\sigma,(P,E).(\varepsilon\backslash E^{\neg})\cup E$ and set $\mathcal{B}:=\mathcal{B}\cup\{\sigma'.\varepsilon':\varphi\}$, else set $\mathcal{B}:=\mathcal{B}\cup\{\sigma'.\varepsilon':\varphi\}$ for some prefix $\sigma'.\varepsilon'$ which is occurring in $\mathcal{B}$ with $\varepsilon'=(\varepsilon\backslash E^{\neg})\cup E$.
$\neg atom_{\langle\rangle}$-rule	If $\sigma.\varepsilon:\neg\langle(P,E)\rangle\varphi\in\mathcal{B}$, $\{\sigma.\varepsilon:\neg\psi\mid\psi\in P\}\cap\mathcal{B}=\varnothing$, and there is some prefix $\sigma'.\varepsilon'$ with $\varepsilon'=(\varepsilon\backslash E^{\neg})\cup E$ and $\sigma'.\varepsilon':\neg\varphi\notin\mathcal{B}$, then set $\mathcal{B}:=\mathcal{B}\cup\{\sigma'.\varepsilon':\neg\varphi\}$, or set $\mathcal{B}:=\mathcal{B}\cup\{\sigma.\varepsilon:\neg\psi\}$ for some $\psi\in P$.

Table 4 Tableau expansion rules on effects of TD_{ALCQIO} actions

表 4 TD_{ALCQIO} 中关于动作效果的扩展规则

Name	Rule
B_1-rule	If $\sigma.\varepsilon:\varphi\in\mathcal{B}$, φ is of the form $R(u,v)$ or $\neg R(u,v)$ for some role R, $\varphi\notin\varepsilon$ and $\sigma_0.\varepsilon_0:\varphi\notin\mathcal{B}$, then set $\mathcal{B}:=\mathcal{B}\cup\{\sigma_0.\varepsilon_0:\varphi\}$.
B_2-rule	If $\sigma.\varepsilon:\varphi\in\mathcal{B}$, φ is of the form $D(u)$ for some concept D, and $\sigma_0.\varepsilon_0:D^{Regress(\sigma.\varepsilon)}(u)\notin\mathcal{B}$, then set $\mathcal{B}:=\mathcal{B}\cup\{\sigma_0.\varepsilon_0:D^{Regress(\sigma.\varepsilon)}(u)\}$.

Table 5 Tableau expansion rules on temporal action assertions of TD_{ALCQIO}

表 5 TD_{ALCQIO} 中关于时序动作断言的扩展规则

Name	Rule
A-rule	If $\sigma.\varepsilon:\psi\in\mathcal{B}$, ψ is of the form $\neg\mathbf{E}(\varphi\mathcal{U}^{\pi}\psi)$, $\neg\mathbf{A}(\varphi\mathcal{U}^{\pi}\psi)$, $\mathbf{E}(\varphi\mathcal{U}^{\pi}\psi)$ or $\mathbf{A}(\varphi\mathcal{U}^{\pi}\psi)$, and no automaton $A=(Q,\Sigma,\delta,q,F)$ exists with both $L(\pi)=L(A(q))$ and $\sigma.\varepsilon:\psi_A\in\mathcal{B}$ (here, be corresponding to the form of ψ, ψ_A denotes a formula of the form $\neg\mathbf{E}(\varphi\mathcal{U}^{A(q)}\psi)$, $\neg\mathbf{A}(\varphi\mathcal{U}^{A(q)}\psi)$, $\mathbf{E}(\varphi\mathcal{U}^{A(q)}\psi)$ or $\mathbf{A}(\varphi\mathcal{U}^{A(q)}\psi)$ respectively), then construct an automaton $A=(Q,\Sigma_{\pi},\delta,q,F)$ with $L(A(q))=L(\pi)$, and set $\mathcal{B}:=\mathcal{B}\cup\{\sigma.\varepsilon:\psi_A\}$.
$A\mathcal{U}$-rule	If $\sigma.\varepsilon:\mathbf{A}(\varphi\mathcal{U}^{A(q)}\psi)\in\mathcal{B}$ for some automaton $A(q)=(Q,\Sigma_{\pi},\delta,q,F)$, $\sigma.\varepsilon:\psi\notin\mathcal{B}$, and either $\sigma.\varepsilon:\varphi\notin\mathcal{B}$ or $\sigma.\varepsilon:\neg(\vee_{\alpha\in\Sigma,q'\in\delta(q,\alpha)}\langle\alpha\rangle\neg\mathbf{A}(\varphi\mathcal{U}^{A(q')}\psi))\notin\mathcal{B}$, then set $\mathcal{B}:=\mathcal{B}\cup\{\sigma.\varepsilon:\psi\}$ or $\mathcal{B}:=\mathcal{B}\cup\{\sigma.\varepsilon:\varphi,\sigma.\varepsilon:\neg(\vee_{\alpha\in\Sigma,q'\in\delta(q,\alpha)}\langle\alpha\rangle\neg\mathbf{A}(\varphi\mathcal{U}^{A(q')}\psi))\}$.
$\neg A\mathcal{U}$-rule	If $\sigma.\varepsilon:\neg\mathbf{A}(\varphi\mathcal{U}^{A(q)}\psi)\in\mathcal{B}$ for some automaton $A(q)=(Q,\Sigma_{\pi},\delta,q,F)$, and either $\sigma.\varepsilon:\neg\psi\notin\mathcal{B}$ or both $\sigma.\varepsilon:\neg\varphi\notin\mathcal{B}$ and $\sigma.\varepsilon:\vee_{\alpha\in\Sigma,q'\in\delta(q,\alpha)}\langle\alpha\rangle\neg\mathbf{A}(\varphi\mathcal{U}^{A(q')}\psi)\notin\mathcal{B}$, then set $\mathcal{B}:=\mathcal{B}\cup\{\sigma.\varepsilon:\neg\psi,\sigma.\varepsilon:\neg\varphi\}$ or $\mathcal{B}:=\mathcal{B}\cup\{\sigma.\varepsilon:\neg\psi,\sigma.\varepsilon:\vee_{\alpha\in\Sigma,q'\in\delta(q,\alpha)}\langle\alpha\rangle\neg\mathbf{A}(\varphi\mathcal{U}^{A(q')}\psi)\}$.
$E\mathcal{U}_f$-rule	If $\sigma.\varepsilon:\mathbf{E}(\varphi\mathcal{U}^{A(q)}\psi)\in\mathcal{B}$ for some automaton $A(q)=(Q,\Sigma_{\pi},\delta,q,F)$, $q\in F$, $\sigma.\varepsilon:\psi\notin\mathcal{B}$, and either $\sigma.\varepsilon:\varphi\notin\mathcal{B}$ or $\sigma.\varepsilon:\vee_{\alpha\in\Sigma,q'\in\delta(q,\alpha)}\langle\alpha\rangle\mathbf{E}(\varphi\mathcal{U}^{A(q')}\psi)\notin\mathcal{B}$, then set $\mathcal{B}:=\mathcal{B}\cup\{\sigma.\varepsilon:\psi\}$ or set $\mathcal{B}:=\mathcal{B}\cup\{\sigma.\varepsilon:\varphi,\sigma.\varepsilon:\vee_{\alpha\in\Sigma,q'\in\delta(q,\alpha)}\langle\alpha\rangle\mathbf{E}(\varphi\mathcal{U}^{A(q')}\psi)\}$.
$\neg E\mathcal{U}_f$-rule	If $\sigma.\varepsilon:\neg\mathbf{E}(\varphi\mathcal{U}^{A(q)}\psi)\in\mathcal{B}$ for some automaton $A(q)=(Q,\Sigma_{\pi},\delta,q,F)$, $q\in F$, and either $\sigma.\varepsilon:\neg\psi\notin\mathcal{B}$ or both $\sigma.\varepsilon:\neg\varphi\notin\mathcal{B}$ and $\sigma.\varepsilon:\neg(\vee_{\alpha\in\Sigma,q'\in\delta(q,\alpha)}\langle\alpha\rangle\mathbf{E}(\varphi\mathcal{U}^{A(q')}\psi))\notin\mathcal{B}$, then set $\mathcal{B}:=\mathcal{B}\cup\{\sigma.\varepsilon:\neg\psi,\sigma.\varepsilon:\neg\varphi\}$ or $\mathcal{B}:=\mathcal{B}\cup\{\sigma.\varepsilon:\neg\psi,\sigma.\varepsilon:\neg(\vee_{\alpha\in\Sigma,q'\in\delta(q,\alpha)}\langle\alpha\rangle\mathbf{E}(\varphi\mathcal{U}^{A(q')}\psi))\}$.
$E\mathcal{U}_{nf}$-rule	If $\sigma.\varepsilon:\mathbf{E}(\varphi\mathcal{U}^{A(q)}\psi)\in\mathcal{B}$ for some automaton $A(q)=(Q,\Sigma_{\pi},\delta,q,F)$, $q\notin F$, and ① $\sigma.\varepsilon:\psi\notin\mathcal{B}$ or $\sigma.\varepsilon:\mathbf{E}(\mathrm{true}\,\mathcal{U}^{A(q)}\mathrm{true})\notin\mathcal{B}$, and ② $\sigma.\varepsilon:\varphi\notin\mathcal{B}$ or $\sigma.\varepsilon:\vee_{\alpha\in\Sigma,q'\in\delta(q,\alpha)}\langle\alpha\rangle\mathbf{E}(\varphi\mathcal{U}^{A(q')}\psi)\notin\mathcal{B}$, then set $\mathcal{B}:=\mathcal{B}\cup\{\sigma.\varepsilon:\psi,\sigma.\varepsilon:\mathbf{E}(\mathrm{true}\,\mathcal{U}^{A(q)}\mathrm{true})\}$ or $\mathcal{B}:=\mathcal{B}\cup\{\sigma.\varepsilon:\varphi,\sigma.\varepsilon:\vee_{\alpha\in\Sigma,q'\in\delta(q,\alpha)}\langle\alpha\rangle\mathbf{E}(\varphi\mathcal{U}^{A(q')}\psi)\}$.
$\neg E\mathcal{U}_{nf}$-rule	If $\sigma.\varepsilon:\neg\mathbf{E}(\varphi\mathcal{U}^{A(q)}\psi)\in\mathcal{B}$ for some automaton $A(q)=(Q,\Sigma_{\pi},\delta,q,F)$, $q\notin F$, and either ① both $\sigma.\varepsilon:\neg\psi\notin\mathcal{B}$ and $\sigma.\varepsilon:\neg\mathbf{E}(\mathrm{true}\,\mathcal{U}^{A(q)}\mathrm{true})\notin\mathcal{B}$, or ② both $\sigma.\varepsilon:\neg\varphi\notin\mathcal{B}$ and $\sigma.\varepsilon:\neg(\vee_{\alpha\in\Sigma,q'\in\delta(q,\alpha)}\langle\alpha\rangle\mathbf{E}(\varphi\mathcal{U}^{A(q')}\psi))\notin\mathcal{B}$, then set $\mathcal{B}:=\mathcal{B}\cup\{\sigma.\varepsilon:\phi_1,\sigma.\varepsilon:\phi_2\}$ for some $\phi_1\in\{\neg\psi,\neg\mathbf{E}(\mathrm{true}\,\mathcal{U}^{A(q)}\mathrm{true})\}$ and some $\phi_2\in\{\neg\varphi,\neg(\vee_{\alpha\in\Sigma,q'\in\delta(q,\alpha)}\langle\alpha\rangle\mathbf{E}(\varphi\mathcal{U}^{A(q')}\psi))\}$.

4 判定算法的性质

本节证明判定算法的可终止性和正确性.

定理 1. 算法 1 是可终止的.

证明:首先引入几个基本符号.对于判定算法中考察的公式φ_{init},用 $size(\varphi_{init})$表示 $nf_{\mathcal{T},\mathcal{A}c}(\varphi_{init})$的长度(即 $\text{TD}_{\text{ALCQIO}}$中的基本符号在$nf_{\mathcal{T},\mathcal{A}c}(\varphi_{init})$中累计出现的次数),用$AtomAct(\varphi_{init})$表示由出现在$nf_{\mathcal{T},\mathcal{A}c}(\varphi_{init})$中并且已经表示为二元组形式的所有原子动作组成的集合.在此基础上,用$Eff(\varphi_{init})$表示集合$\bigcup_{(P_i,E_i)\in AtomAct(\varphi_{init})} E_i$.令$e:=\#Eff(\varphi_{init})$,显然,$e$ 与 $size(\varphi_{init})$呈线性关系.

接下来,算法 1 的可终止性基于以下结论:

(1) 每应用一次表 1~表 5 中的某条扩展规则之后,分枝 $\mathcal{B}$ 的元素个数都至少增加 1;

(2) 根据 $\text{atom}_{\langle\rangle}$-规则,对于 $\mathcal{B}$ 中出现的任一前缀$\sigma.\varepsilon$,都有$\varepsilon\subseteq E$,并且对于 $\mathcal{B}$ 中出现的任意两个前缀$\sigma.\varepsilon$和$\sigma'.\varepsilon'$,都满足$\varepsilon\neq\varepsilon'$.因此,$\mathcal{B}$中出现的前缀个数最多为 2^e-1;

(3) 对于每个时序动作断言$\psi\mathcal{U}^{\pi}\psi$,应用 A-规则后将引入某个自动机 $A=(Q,\Sigma,\delta,q,F)$,其中,A 的状态数量与π的长度(即原子动作、测试动作以及动作构造符“∪”,“;”和“*”在π中累计出现的次数)呈线性关系;在此基础上,进一步应用表 5 中其他扩展规则之后,新引入的形如$\psi\mathcal{U}^{A(q')}\psi$ 或$\neg(\psi\mathcal{U}^{A(q')}\psi)$(其中,$q'\in Q$)的公式个数与$A$的状态数量呈线性关系.此外,令$\psi_1\mathcal{U}^{\pi_1}\psi_1,\ldots,\psi_m\mathcal{U}^{\pi_m}\psi_m$为$\varphi_{init}$中出现的所有时序动作断言,则$\pi_1,\ldots,\pi_m$的长度之和必然小于 $size(\varphi_{init})$.因此,应用表 5 中各条扩展规则之后所能引入的所有带前缀的公式数量与 $size(\varphi_{init})$呈线性关系;

(4) 表 1~表 4 中的扩展规则与文献[7]中的扩展规则相似.与文献[7]的证明相似,应用这些扩展规则之后可能引入的所有带前缀的公式、关于个体名的等式以及关于个体名的不等式都是有限的. □

为了证明算法1的正确性,我们先引入分支-结构映射,将分支中出现的前缀映射到结构中的可能世界;在此基础上引入分枝的可满足性问题,并将其与公式的可满足性问题对应起来.

定义 15. 令Σ_{prefix}是由出现在 $\mathcal{B}$ 中的所有前缀组成的集合,$M=(\Sigma,\Delta_M,I)$为 $\text{TD}_{\text{ALCQIO}}$ 结构;将某个函数δ: $\Sigma_{prefix}\rightarrow W$称为从$\mathcal{B}$到$M$的分支-结构映射,当且仅当对于$\Sigma_{prefix}$中的任意两个前缀$\sigma.\varepsilon,\sigma'.\varepsilon'$来说:如果存在原子动作$(P,E)$使得$\sigma'=\sigma;(P,E)$和$\varepsilon'=(\varepsilon\backslash E^{\neg})\cup E$,则必然有:

(1) 对于出现在$\mathcal{B}$中的任一简单概念名 C 都有 $C^{I(\delta(\sigma'.\varepsilon'))}=(C^{I(\delta(\sigma.\varepsilon))}\cup\{u^I|C(u)\in E\})\backslash\{u^I|\neg C(u)\in E\}$;

(2) 对于出现在$\mathcal{B}$中的任一角色名 R 都有 $R^{I(\delta(\sigma'.\varepsilon'))}=(R^{I(\delta(\sigma.\varepsilon))}\cup\{(u^I,v^I)|R(u,v)\in E\})\backslash\{(u^I,v^I)|\neg R(u,v)\in E\}$.

定义 16. 令$\mathcal{B}$为任一分支,$M=(\Sigma,\Delta_M,I)$为 $\text{TD}_{\text{ALCQIO}}$结构,$\delta$为从$\mathcal{B}$到$M$的分支-结构映射.称结构$M$和映射$\delta$满足分支$\mathcal{B}$,表示为$M,\delta\models\mathcal{B}$,当且仅当:① 对于$\mathcal{B}$中任一带前缀的公式$\sigma.\varepsilon:\varphi$,都有$(M,\delta(\sigma.\varepsilon))\models\varphi$;② 对于$\mathcal{B}$中任一关于个体名的等式$u=v$,都有$u^I=v^I$;以及③ 对于$\mathcal{B}$中任一关于个体名的不等式$u\neq v$,都有$u^I\neq v^I$.

对于分支$\mathcal{B}$,如果存在某个结构M以及相应的分支-结构映射δ使得$M,\delta\models\mathcal{B}$,则称$\mathcal{B}$是可满足的;否则,称$\mathcal{B}$为不可满足的.

基于上述定义,可以证明算法 1 具有以下性质:

引理 1. 对于在分支$\mathcal{B}$上应用任意一条扩展规则的过程来说,$\mathcal{B}$在扩展之前是可满足的当且仅当可以在扩展后得到某个可满足的分支.

引理 2. 如果分支$\mathcal{B}$是饱和的、无冲突的并且是不可忽略的,则$\mathcal{B}$一定是可满足的.

引理 3. 如果分支$\mathcal{B}$存在冲突,则$\mathcal{B}$一定是不可满足的.

引理 4. 如果算法 1 构造的初始分支 $\mathcal{B}_{init}:=\{\sigma_0.\varepsilon_0:nf_{\mathcal{T},\mathcal{A}c}(\varphi_{init})\}$是可满足的,则在应用扩展规则后得到的所有饱和的并且可满足的分支中,必然存在某个分支是不可忽略的.

其中,对表 1~表 5 的扩展规则进行归纳证明后容易得到引理 1;对 $\text{TD}_{\text{ALCQIO}}$中公式的结构归纳证明后容易得到引理 2;根据对冲突的定义可以直接得到引理 3.对引理 4 的证明可以采用反证法,假设应用扩展规则后得到的各个饱和且可满足的分支都是可忽略的;虽然每个可忽略的分支可能是由不同的未被实现的可能性标记或

者不同的未被实现的带自动机的存在性时序动作断言所导致,但不失一般性,我们只需考察某个未被实现的可能性标记 $X\equiv\langle\pi^*\rangle\varphi$以及某个未被实现的带自动机的存在性时序动作断言$\sigma.\varepsilon.\boldsymbol{E}(\varphi\mathcal{U}^{A(q)}\psi)$.对这些引理的详细证明过程与文献[6,7]中类似;限于篇幅,这里不再详述.

最后,根据定理 1 以及引理 1~引理 4,容易证明算法 1 的正确性,即:

定理 2. 算法 1 返回结果"φ_{init}相对于 $\mathcal{T}$和 $\mathcal{A}c$ 是可满足的"当且仅当φ_{init}相对于 $\mathcal{T}$和 $\mathcal{A}c$ 是可满足的.

由算法 1 的可终止性和正确性可进一步得到如下结论:

推论 1. $\mathrm{TD}_{\mathrm{ALCQIO}}$中公式的可满足性问题是可判定的.

5 相关工作

目前,比较成熟的动作理论可以分为两大类:基于谓词逻辑的动作理论与采用命题语言的动作理论.这两类动作理论之间存在一个关于描述能力和推理能力(或推理性能)的鸿沟:前者描述能力很强但相关推理问题为不可判定,后者可判定但描述能力局限在命题层面上.近年来,针对这个鸿沟,国内外相关研究者提出了基于描述逻辑的动作理论.

Baader 等人[15]首先构建了一个基于描述逻辑 ALCQIO 的动作形式系统.在该系统中,由概念定义式组成的 TBox 被用于刻画动作理论的背景信息;由个体断言组成的 ABox 被用于刻画世界状态;每个原子动作被刻画为(*pre*,*occ*,*post*)的三元组形式,分别表示执行动作之前必须满足的前提条件、执行过程中的不确定成分以及执行后产生的影响.其中的 *pre*,*occ* 和 *post* 都用描述逻辑中的个体断言来刻画,并且 *occ* 和 *post* 中出现的必须是相对于 TBox 的简单公式.原子动作的有限序列构成复杂动作.在此基础上,该文将动作的可执行性问题和投影问题转换为描述逻辑中 ABox 的一致性问题来解决.Baader 等人对该动作系统进行了许多限制,不允许测试、选择、迭代等具有复杂结构的动作.

Gu 和 Soutchanski[16]以描述逻辑为参照,提出对情景演算的一种改造形式.一方面,在基于情景演算的动作理论的基础上,将刻画动作以及初始情景的公式都限制为 C^2 逻辑中的公式;另一方面,将描述逻辑中的 TBox 和 RBox 引入到情景演算中.最终得到的动作系统既具有较强的描述能力,又保证了相关推理问题是可判定的.在此基础上,该文对动作的可执行性问题和投影问题进行了深入考察.

上述两项工作的基本思路都是从情景演算出发,以描述逻辑的研究成果作为参照,对情景演算进行限定,使得最终得到的动作系统在具有较强描述能力的同时又保证了相关推理问题的可判定性.Gu 和 Soutchanski 给出的动作系统本质上都是情景演算的特殊形式.这种途径使得关于情景演算的许多研究成果和工具都可以作为参考或直接使用,但另一方面也不可避免地继承了情景演算的某些局限.例如,相关推理问题仍然要转换为一阶逻辑的定理证明问题.此外,情景演算中应用 GOLOG 或直接应用 Prolog 进行规划时,Prolog 所基于的闭世界假设要求所输入的信息是完全的;对于信息不完全的情况,需要借助定理证明器进行额外的处理等.

与上述研究工作不同,本文遵循了文献[5−7]的基本思路,将动态逻辑和动作理论与描述逻辑结合起来构建逻辑系统,进而形成基于描述逻辑的动作理论.然而,与文献[5−7]不同,本文进一步引入了分支时序逻辑的刻画成分,可以对整个动态系统的可达性和安全性等时序特征进行刻画和推理.这种对时序特征的刻画和推理是以上各种基于描述逻辑的动作理论都不支持的.

Henriksen 等人[10]将命题动态逻辑与线性时序逻辑结合后提出了动态线性时序逻辑 DLTL;之后,Giordano 等人基于自动机技术给出了复杂度为 PSPACE 的关于 DLTL 的判定算法[17],并将 DLTL 应用于对动作的表示和推理[18]以及对交互协议的刻画和验证等[19].本文借鉴了 DLTL 中对时间的处理方式,将时间的进展体现为原子行动的执行.但是,DLTL 是基于线性时序结构的,而 $\mathrm{TD}_{\mathrm{ALCQIO}}$是基于分支时序结构的.更为关键的是,与 DLTL 相比,$\mathrm{TD}_{\mathrm{ALCQIO}}$进一步结合了描述逻辑以及基于描述逻辑的动作理论.虽然基于 DLTL 也可以构建动作理论[18],但其中对世界状态、动作的前提条件、动作所能产生的效果等等的刻画只能采用命题公式;而 $\mathrm{TD}_{\mathrm{ALCQIO}}$ 支持的是基于描述逻辑的动作理论,其中采用了描述逻辑公式对世界状态和动作等进行更细致的刻画,具有更强的描述能力.

6 结 论

时序动态描述逻辑 TD_{ALCQIO} 将描述逻辑 ALCQIO、基于描述逻辑的动作理论以及动态逻辑和分支时序逻辑等有机地结合了起来;除了可以基于描述逻辑 ALCQIO 对静态领域知识以及关于动作的知识进行刻画以外,还可以对整个动态领域的动态特征和时序特征进行刻画和推理,并且在推理时保持了可判定性特征.与其他基于描述逻辑的动作理论相似,TD_{ALCQIO} 提供了一种对语义 Web 服务进行建模和推理的途径.在此基础上,TD_{ALCQIO} 关于动态特征和时序特征的刻画及推理能力为 Web 服务的发现、组合和验证等提供了进一步的支持.

我们下一步的工作是分析 TD_{ALCQIO} 中公式可满足性问题的复杂度下限,并参照分析结果对本文的 Tableau 算法进行优化和实现.另一项工作是从模型检测的角度对 TD_{ALCQIO} 进行研究,围绕 TD_{ALCQIO} 的应用将逻辑推理途径与模型检测途径很好地结合起来.

致谢 感谢审稿专家提出的宝贵意见.

References:

[1] Bonatti P, Lutz C, Wolter F. Description logics with circumscription. In: Doherty P, Mylopoulos J, Welty C, eds. Proc. of the 10th Int’l Conf. on Principles of Knowledge Representation and Reasoning. Menlo Park: AAAI Press, 2006. 400−410.

[2] Ma Y, Hitzler P, Lin ZQ. Algorithms for paraconsistent reasoning with OWL. In: Franconi E, Kifer M, May W, eds. Proc. of the 4th European Semantic Web Conf. Belin: Springer-Verlag, 2007. 399−413. [doi: 10.1007/978-3-540-72667-8_29]

[3] Kang DZ, Xu BW, Lu JJ, Li YH. Reasoning within extended fuzzy description logic supporting terminological axiom restrictions. Journal of Software, 2007,18(7):1563−1572 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/18/1563.htm [doi: 10.1360/jos181563]

[4] Jiang YC, Wang J, Tang SQ, Xiao B. Reasoning with rough description logics: An approximate concepts approach. Information Sciences, 2009,179(5):600−612. [doi: 10.1016/j.ins.2008.10.021]

[5] Shi ZZ, Dong MK, Jiang YC, Zhang HJ. A logical foundation for the semantic Web. Science in China (Series F: Information Sciences), 2005,48(2):161−178. [doi: 10.1360/03yf0506]

[6] Chang L, Shi ZZ, Qiu LR, Lin F. A tableau decision algorithm for dynamic description logic. Chinese Journal of Computers, 2008, 31(6):896−909 (in Chinese with English abstract).

[7] Chang L, Shi ZZ, Chen LM, Niu WJ. Family of extended dynamic description logics. Journal of Software, 2010,21(1):1−13 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/21/1.htm [doi: 10.3724/SP.J.1001.2010.03494]

[8] Shi ZZ, Chang L. Reasoning about semantic Web services with an approach based on dynamic description logics. Chinese Journal of Computers, 2008,31(9):1599−1611 (in Chinese with English abstract).

[9] Gabbay DM, Kurucz A, Wolter F, Zakharyaschev M. Many-Dimensional Modal Logics: Theory and Applications. New York: Elsevier, 2003. 367−375.

[10] Henriksen JG, Thiagarajan PS. Dynamic linear time temporal logic. Annals of Pure and Applied Logic, 1999,96(1-3):187−207. [doi: 10.1016/S0168-0072(98)00039-6]

[11] Clarke EM, Emerson EA. Design and synthesis of synchronization skeletons using branching time temporal logic. In: Kozen D, ed. Proc. of the Logic of Programs. LNCS 131, Heidelberg: Springer-Verlag, 1982. 52−71. [doi: 10.1007/BFb0025774]

[12] Chang L, Lin F, Shi ZZ. A dynamic description logic for representation and reasoning about actions. In: Zhang ZL, Siekmann J, eds. Proc. of the 2nd Int’l Conf. on Knowledge Science, Engineering and Management. Berlin: Springer-Verlag, 2007. 115−127. [doi: 10.1007/978-3-540-76719-0_15]

[13] Wang J, Jiang YC, Shen YM. Satisfiability and reasoning mechanism of terminological cycles in description logic *vL*. Science in China (Series F: Information Sciences), 2008,51(9):1204−1214. [doi: 10.1007/s11432-008-0101-6]

[14] Hromkovic J, Seibert S. Translating regular expressions into small ε-free nondeterministic finite automata. Journal of Computer and System Sciences, 2001,62(4):565−588. [doi: 10.1006/jcss.2001.1748]

[15] Baader F, Lutz C, Miličić M, Sattler U, Wolter F. Integrating description logics and action formalisms: First results. In: Veloso M, Kambhampati S, eds. Proc. of the 12th National Conf. on Artificial Intelligence. Menlo Park: AAAI Press/The MIT Press, 2005. 572−577.

[16] Gu YL, Soutchanski M. Decidable reasoning in a modified situation calculus. In: Veloso M, ed. Proc. of the 20th Int’l Joint Conf. on Artificial Intelligence. Menlo Park: AAAI Press, 2007. 1891−1897.

[17] Giordano L, Martelli A. Tableau-Based automata construction for dynamic linear time temporal logic. Annals of Mathematics and Artificial Intelligence, 2006,46(3):289−315. [doi: 10.1007/s10472-006-9020-7]

[18] Giordano L, Martelli A, Schwind C. Reasoning about actions in dynamic linear time temporal logic. Logic Journal of the IGPL, 2001,9(2):273−288. [doi: 10.1093/jigpal/9.2.273]

[19] Giordano L, Martelli A, Schwind C. Specifying and verifying interaction protocols in a temporal action logic. Journal of Applied Logic, 2007,5(2):214−234. [doi: 10.1016/j.jal.2005.12.011]

附中文参考文献:

[3] 康达周,徐宝文,陆建江,李言辉.支持术语公理约束的扩展模糊描述逻辑推理.软件学报,2007,18(7):1563−1572. http://www.jos.org.cn/1000-9825/18/1563.htm [doi: 10.1360/jos181563]

[6] 常亮,史忠植,邱莉榕,林芬.动态描述逻辑的 Tableau 判定算法.计算机学报,2008,31(6):896−909.

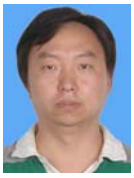
[7] 常亮,史忠植,陈立民,牛温佳.一类扩展的动态描述逻辑.软件学报,2010,21(1):1−13. http://www.jos.org.cn/1000-9825/21/1.htm [doi: 10.3724/SP.J.1001.2010.03494]

[8] 史忠植,常亮.基于动态描述逻辑的语义 Web 服务推理.计算机学报,2008,31(9):1599−1611.

常亮(1980－),男,贵州赫章人,博士,副教授,CCF 会员,主要研究领域为知识表示与推理,智能规划,形式化方法.

古天龙(1964－),男,博士,教授,博士生导师,主要研究领域为形式化方法,符号计算.

史忠植(1941－),男,研究员,博士生导师,CCF 高级会员,主要研究领域为人工智能,智能主体,机器学习.



王晓峰(1978－),男,博士,CCF 学生会员,主要研究领域为描述逻辑,数据挖掘,机器学习.